

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-27

ANÁLISIS A LAS VENTAJAS Y DESVENTAJAS DE LOS ESCENARIOS EN LA ELICITACIÓN DE REQUISITOS

Ángela María Tascon Calvo

Francisco Orlando Domínguez H.

FACULTAD DE INGENIERÍAS

Ingeniería de Sistemas

Director
Prof. Edgar Serna M.

INSTITUTO TECNOLÓGICO METROPOLITANO

Febrero 2016

RESUMEN

Debido a la creciente complejidad de los problemas que se pueden resolver mediante productos software, el desarrollo de este producto tecnológico ha tenido que innovar en técnicas y herramientas para responder a las exigencias de calidad y fiabilidad que la sociedad exige. Además, dado que la Ingeniería de Requisitos es considerada la fase más importante de su ciclo de vida, es necesario conocer cómo se está llevando a cabo para que las instituciones de educación tengan en cuenta estos contextos en sus planes de estudios. Conocer las ventajas y desventajas de la técnica de los escenarios para elicitar requisitos brinda la posibilidad de mejorar cada día la forma como se especifican y se aporta al mejoramiento de la calidad que espera del software. De esta forma la sociedad podrá recibir productos que acerquen de mejor forma a la solución de sus problemas.

Palabras clave: Ingeniería de requisitos, técnicas de elicitación, calidad del software.

RECONOCIMIENTOS

Este trabajo de investigación es el resultado del esfuerzo, dedicación y paciencia de los diferentes actores que cruzan a lo largo de nuestra carrera profesional, todos indistintamente, en mucho o poco aportan a nuestro crecimiento y formación de muchos aspectos de la vida, ahora queremos hacer mención de algunos que fueron pilares para este trabajo.

Queremos agradecer muy especialmente a nuestro asesor y Director de Proyecto de investigación el profesor Edgar Serna, por su entrega, paciencia, constancia, apoyo extraordinario, calidad humana y dedicación, durante este largo periodo en el que duró la elaboración total de este artículo, menciones también para los directivos y docentes de la Institución Universitaria (ITM), que se involucraron en todo el trasegar de nuestra formación académica mejorando cada día nuestra idea de un profesional integro.

No podemos olvidar y dejar de mencionar a nuestros padres, son las personas incondicionales que nos brindan su cariño y apoyo durante los momentos buenos y malos de nuestras vidas, tan solo con una palabra o una voz de aliento con las que siempre inspiran a la superación y el mejoramiento constate de nuestro futuro.

Para todos aquellos que directa o indirectamente, nos encontramos y ayudaron para forjar todo este proceso, nuestro mayor reconocimiento y gratitud.

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	5
2. MARCO TEÓRICO	7
3. METODOLOGÍA	9
4. RESULTADOS Y DISCUSIÓN	11
5. CONCLUSIONES	19
REFERENCIAS	21
APÉNDICE (Artículo científico).....	25

1. INTRODUCCION

La elicitación de requisitos es una etapa de la Ingeniería de Requisitos en la que se descubren las necesidades que los usuarios requieren satisfacer con un sistema (Serna y Serna, 2015). En el proceso se utilizan diversas técnicas, tales como entrevistas, encuestas, observación, grupos focales, talleres, cuestionarios, análisis de documentos y reuniones creativas, en las que es necesaria una buena comunicación entre clientes, usuarios, ingenieros y otros que tengan algún interés en el desarrollo del sistema. Una de los problemas en esta fase es que muchas de esas necesidades, empresariales o técnicas, no están documentadas, sino que residen en la mente de los interesados, en la retroalimentación que aún no se ha obtenido de los usuarios finales y en los estudios de diagramas de procesos y encuestas que aún no se han creado. De esta manera, los requisitos deben ser elicitados, o extraídos, y la metodología seleccionada para hacerlo debe ser lógica y meticulosa.

Llevar a cabo una elicitación efectiva es importante y no se puede tomar a la ligera, porque la información obtenida en esta etapa es la base para la especificación de requisitos, que a su vez es la base para las demás fases del ciclo de vida del producto. Se ha demostrado que los errores cometidos en la elicitación suelen ser las principales causas del fracaso o el abandono de los proyectos y que tienen un costo muy alto (Zhang, 2007; Serna, 2012). De ahí la importancia de la elicitación, aunque mayoría de las herramientas de requisitos no la atienden específicamente. Y puede ser aún más complicado, cuando en el desarrollo de muchas aplicaciones el usuario no está disponible para entrevistas o reuniones de grupos focales. Si bien muchos de los enfoques de desarrollo tradicionales son aplicables a la mayoría de proyectos, el análisis de requisitos plantea retos especiales cuando los usuarios son anónimos o no están físicamente disponibles para participar en la elicitación. Esta situación se complica por la diversidad potencial que puede existir entre usuarios y clientes. Entonces, ¿cómo capturar los requisitos de los usuarios, cuando sus aportes son voluntarios, no son de fácil acceso y no existen incentivos suficientes para que participe en la elicitación?

Desde la última década del siglo XX se inició el desarrollo de un método que diversos autores recomiendan para salvar estas dificultades: los escenarios. Este método consiste en utilizar personajes para definir características y atributos de los distintos tipos de usuarios. Se documentan los supuestos clave acerca de ellos, tales como el nivel de conocimientos informáticos y del negocio, las habilidades, las horas de trabajo, las expectativas, y otros similares. El uso de escenarios se ha convertido en una buena práctica para la elicitación de requisitos, al punto que algunos ya la consideran una técnica efectiva en la ausencia física de las partes interesadas (Gottesdiener, 2008). Los escenarios se consideran un medio práctico y útil en la elicitación de requisitos, y Sutcliffe (2003) afirma que son ejemplos de sesiones de interacción que consisten en descripciones de acciones secuenciales, que se relacionan con ejemplos de la vida real, en lugar de descripciones abstractas de las funciones del producto. Las relaciones entre los usuarios y los sistemas se dan fácilmente a través de los escenarios, como una declaración abstracta de lo que éstos requieren de ellos. Además, son útiles para añadir información al entorno de la descripción del requisito. En general, un escenario contiene información no solamente sobre el funcionamiento del sistema, sino también sobre su entorno. Un escenario puede contener la descripción de las acciones en los sistemas manuales, descripciones de los actores, los roles y también de la organización. El interés de la investigación que se presenta en este artículo radica en la comprensión de la utilidad de la técnica de escenarios cuando se aplica a diferentes contextos y proyectos en la elicitación de requisitos. Para eso fue necesario analizar las ventajas y desventajas que se reportan en la literatura y conocer las experiencias que la comunidad ha logrado en su aplicación. Se presenta un análisis a la relación, la importancia, las ventajas y las desventajas de utilizar los escenarios en la elicitación de requisitos. Se hace una descripción de los trabajos relacionados, luego se detalla el estado del arte en relación con la elicitación de requisitos y los escenarios. Se describe la metodología de la investigación; se presenta un análisis de los beneficios y problemas de los escenarios en la elicitación, y finalmente se concluye el trabajo realizado.

2. MARCO TEORICO

Un proyecto de ingeniería de software moderno implica la participación de un grupo diverso de diferentes partes interesadas, cuyas necesidades y expectativas, además de, por ejemplo, las leyes y normas de la industria constituyen la base de los requisitos del sistema software. Para completar con éxito un proyecto de ingeniería de software se debe alcanzar un acuerdo en las primeras etapas del proyecto sobre qué capacidades deben ser entregadas y qué más se requiere del sistema (Hruschka, 1997).

Uno de los problemas más difíciles en el proceso de decidir los requisitos del sistema es la falta de comunicación entre las diferentes partes interesadas. Los analistas de sistemas tienden a hablar con términos técnicos, mientras que el cliente y los usuarios finales prefieren utilizar el lenguaje cotidiano que utilizan en sus actividades diarias. Como resultado, los primeros pueden no tener una idea clara de qué tipo de sistema están construyendo y el cliente y los usuarios finales a menudo se sorprenden cuando se dan cuenta que el producto final no cumple con sus expectativas ni resuelve los problemas que tenían al comienzo. Una solución conocida para reducir la brecha de comunicación es el uso de prototipos, que son algo concreto que presentan las necesidades de las partes interesadas (Kimmond, 1995). Otra solución más reciente para mejorar la calidad de la comunicación en la ingeniería de requisitos, y por tanto del software, es la introducción de casos de uso y escenarios, que permiten a los diferentes actores describir y revisar el problema en su propio lenguaje en lugar de algún modelo abstracto (Weidenhaupt et al., 1998).

Los escenarios son comunes en la ingeniería de requisitos y existen diversos métodos para su uso (Jacobson, 1992; Kulak y Guiney, 2000; Schneider y Winters, 2001), y también existen una serie de métodos para el uso de prototipos (Asur y Hufnagel, 1993; Leffingwell y Widrig, 1999; Sommerville y Sawyer, 1997). Sin embargo, a pesar de que los casos de uso y los prototipos se utilizan con frecuencia juntos en el desarrollo de software, existen pocos enfoques prácticos que combinen la especificación de requisitos, los escenarios, los prototipos y el desarrollo del sistema evolutivo (Weidenhaupt, 1998). Para hacer que la participación de los usuarios finales en el

desarrollo de los requisitos sea más natural es necesario tener un enfoque que le ayude al usuario a crear una imagen del problema en el contexto de uso real. Por lo tanto, es necesario desarrollar métodos sistemáticos que permitan conocer las ventajas y desventajas de esta técnica de elicitación de requisitos.

3. METODOLOGIA

Una revisión de la literatura debe ser completa y centrada en conceptos. Por lo tanto, dos de las cuestiones relevantes que se deben resolver son cómo identificar la literatura relevante y cómo estructurar el análisis de los hallazgos (Weill y Olson, 1989; Webster y Watson, 2002; Zhang y Li, 2005). Para resolver la primera cuestión, en el desarrollo de la investigación se consideraron las publicaciones relacionadas con el desarrollo de Sistemas de Información, la Ingeniería de Requisitos y las técnicas para elicitar requisitos. Estas corrientes de la literatura ofrecen perspectivas relevantes para cumplir con el objetivo respectivo, primero, desde un punto de vista especializado (Davis, 1982) y segundo desde un punto de vista más general (McFarlan, 1981, 1982). Webster y Watson (2002) sugieren identificar artículos relevantes en las principales revistas y luego revisar las citas que se hacen de ellos; y Weill y Olson (1989) sugieren aplicar la crítica estructurada para guiar la selección de artículos adicionales. Se utilizó el servicio Web of Science con acceso a la literatura científica para identificar los trabajos y las investigaciones pertinentes. Se utilizaron palabras clave generales y se identificaron 500 artículos relevantes en los temas de búsqueda. Posteriormente, se seleccionaron los artículos publicados en las revistas mejor clasificadas. En este proceso se definieron 113 artículos, pero muchos resultaron tratar la temática de forma periférica o sin relevancia para el estudio. Por ello, se realizó un filtro manual con base en el criterio de que debían evaluar la técnica de escenarios para elicitar requisitos, y se eliminaron 23 trabajos. Luego se revisaron las citas que se hicieron de ellos. En este proceso se compilaron los artículos que tenían tres o más citas en las principales revistas, con lo que la muestra final quedó conformada por 63 trabajos.

Para resolver la segunda cuestión, cómo estructurar el análisis de los hallazgos, se aplicó la teoría de contingencia con el apoyo del modelo unificado de elicitación de requisitos (Hickey y Davis, 2004), que sugiere utilizar factores contextuales como base para adoptar una técnicas de requisitos. Para superar las limitaciones de la teoría de contingencia clásica, se adoptó la gestión del riesgo como una forma específica de la

Teoría que se ha aplicado con éxito en la disciplina de Sistemas de Información (Lyytinen, 1988; Lyytinen, Mathiassen y Ropponen, 1998). Los modelos de contingencia y los enfoques de gestión de riesgos se basan en visiones particulares sobre cómo responder efectivamente en contextos específicos. Es por ello que se analiza la literatura identificada para entender cómo aplicar patrones de técnicas de elicitación de requisitos en arquetipos de perfiles de riesgo. El objetivo era determinar, mediante el riesgo y la contingencia, las ventajas y desventajas de las técnicas de elicitación presentadas en la literatura, especialmente sobre la técnica de escenarios.

CRONOGRAMA DE ACTIVIDADES

Metas	1	2	3	4	5	6	7
1. Construir protocolo de búsqueda	X						
2. Realizar revisión de la literatura	X	X	X				
3. Definir criterios de análisis			X	X			
4. Determinar el nivel de aplicación y encontrar dificultades y problemas				X	X		
5. Aplicar métrica de análisis comparativo					X	X	
6. Redactar informe para artículo				X	X	X	X

4. RESULTADOS Y DISCUSION

A mediados del siglo XX, los requisitos para las primeras aplicaciones software eran fáciles de identificar, porque la mayoría eran desarrolladas por los científicos para apoyar sus propias necesidades y propósitos. Sin embargo, a medida que las empresas comenzaron a desarrollar software para los usuarios, pronto se hizo necesario reunir, explicar y entender las necesidades de los usuarios, y que participaran en el proceso de desarrollo del producto (McKeen, Guimaraes y Wetherbe, 1994; Lamb y Kling, 2003; Markus y Mao, 2004). Esto ha dado lugar al surgimiento de una considerable variedad de técnicas para apoyar la elicitación de esas necesidades (Davis, 1982; Byrd, Cossick y Zmud, 1992; Keil y Carmel, 1995; Chatzoglou y Macaulay, 1996). Algunos argumentan que el flujo constante de técnicas se ha convertido en una jungla metodológica (Jayaratna, 1994).

Buscando una solución, los investigadores han respondido proponiendo modelos de contingencia para la adopción de estas técnicas a contextos específicos. Algunos ofrecen categorías de factores contextuales, portafolios de técnicas y enfoques para adaptarlas a cada contexto (Kickert, 1983; Iivari, 1992); otros integran conocimiento dentro de determinadas disciplinas (Barki, Rivard y Talbot, 2001; Andrea y Zmud, 2002; Hickey y Davis, 2004) y a menudo se basan en modelos de gestión de riesgos (Lyytinen, Mathiassen y Ropponen, 1998).

Unos de los primeros intentos por integrar el conocimiento disponible para elicitación de requisitos, Alter y Ginzberg (1978) propusieron una técnica de contingencia para el soporte de decisiones; McFarlan (1981, 1982) sugirió el desarrollo de aplicaciones software a través de la integración interna entre los desarrolladores y externa entre desarrolladores y usuarios; y Davis (1982) se centró en el desarrollo de las necesidades de los usuarios integrando el conocimiento disponible mediante una técnica contingente para reducir su incertidumbre.

Por desgracia, y a pesar que la práctica y la investigación han cambiado considerablemente en los últimos años (Neill y Laplante, 2003), no ha habido intentos recientes para integrar los nuevos conocimientos disponibles acerca de la elicitación de

requisitos (Hickey y Davis, 2004). Desde hace algún tiempo, la técnica de escenarios ha penetrado con énfasis en las aplicaciones entre organizaciones, y las nuevas demandas han introducido nuevas técnicas que han cambiado el perfil de la Ingeniería de Requisitos. Debido a la complejidad de los problemas actuales, los equipos de requisitos a menudo se enfrentan al reto de establecer una interacción efectiva con usuarios que están por fuera del alcance organizacional (Frolick y Robichaux, 1995; Peffers, Gengler y Tuunanen, 2003; Duggan y Thachenkary, 2004), un reto que se incrementa cuando los usuarios no saben cómo describir sus necesidades (Walz et al., 1993).

En cuanto al enfoque del modelo de escenarios, Hooper y Hsiao (1982) fueron los primeros en proponer su uso en la Ingeniería de Requisitos. Desarrollaron una metodología para identificar requisitos mediante escenarios, que sirven como un prototipo inicial rápido del sistema previsto. Chen et al. (1994) volvieron a examinar esta idea en su trabajo, y exploraron de manera sistemática cómo usar escenarios en el análisis de requisitos. Utilizaron una base matemática formal y generaron escenarios precisos, con capacidad para el cambio, y manteniendo a los usuarios involucrados en el proceso. Kyng (1995a) jugó un papel decisivo en la elaboración de la relación entre los escenarios y los prototipos. Su investigación en diseño cooperativo enfatizó el uso de herramientas de baja tecnología, como escenarios para apoyar el diseño. En su propuesta, los escenarios se desarrollan para los usuarios finales como un prototipo que contribuye al diseño en curso.

La primera referencia a la utilización de escenarios específicamente para elicitación de requisitos la presentó Holbrook (1990), quien describe una metodología que utiliza principios de la interacción entre los usuarios y diseñadores para desarrollar rápidamente un conjunto de requisitos iniciales usando escenarios. Su investigación proporciona una visión general de cómo utilizar los escenarios para refinar los objetivos y establecer los requisitos. Antón, Potts y Takahashi (1994a) proponen un modelo en el que describen cómo representar escenarios como ejecuciones de planes dirigidos a objetivos, proporcionando un puente entre los requisitos en la fase de planificación con los de la fase de análisis. Asimismo, exploraron aún más el modelo como una estructura formal y propusieron actividades como: elicitación, documentación y refinamiento (Antón, Potts y Takahashi, 1994b). Posteriormente,

Potts (1995) hizo una conexión directa entre las historias narrativas y el modelado de escenarios, y propuso una forma de aplicar ideas narrativas a las necesidades del usuario. Propuso directrices para definir los escenarios más destacados y cómo podrían ser utilizados en un sistema interactivo. Además, describe un método para escribir y utilizar escenarios para analizar las necesidades de los usuarios y del sistema.

Para finales del siglo XX había una multitud de métodos para utilizar escenarios en todo el proceso de la Ingeniería de Requisitos. En un enciclopédico trabajo, Pohl (1997) propuso un marco global para la caracterización del proceso de requisitos: los cuatro mundos del desarrollo, el asunto, el sistema y el uso; y tres dimensiones que incluyen acuerdo, representación y especificación. Los investigadores más importantes en el tema de requisitos colaboraron en el desarrollo de un marco para organizar todos los métodos de escenarios de entonces en un cuerpo unificado y coherente de trabajo. Achour et al. (1996) exploraron las cuestiones subyacentes a los enfoques basados en escenarios para la elicitación de requisitos, y propone un marco para clasificarlos a partir de puntos de vista: 1) del formulario, del contenido, del propósito y del ciclo de vida; 2) un conjunto de facetas asociado que caracteriza y clasifica un escenario; y 3) estas facetas son medidas mediante un conjunto de atributos relevantes. En este mismo sentido, la propuesta de Alexander y Maiden (2004) crea un marco para clasificar los enfoques de escenarios y para comunicar la investigación y el desarrollo del trabajo acerca de los sistemas basados en escenarios.

Davis y Hickey (2003) introdujeron un nuevo modelo unificado de elicitación de requisitos que mostraba el papel crítico de la selección del conocimiento y de la técnica. Alexander y Maiden (2004) trabajaron sobre la idea de que la complejidad de los sistemas se puede manejar si las necesidades de diseño se definen temprana y cuidadosamente. Propusieron a los escenarios como de las técnicas más poderosas para descubrir y comunicar los requisitos. Carroll and Go (2004) sugirieron unificar todos los métodos de planificación de escenarios en una jerarquía con base en las comunidades que emplean este método de diseño: la planificación estratégica, la Ingeniería de Requisitos, el diseño de interfaz humano-computador y el diseño orientado a objetos. Cheng y Atlee (2007) revisaron la investigación en Ingeniería de Requisitos durante la última década del siglo XX y se concentraron en la identificación

de líneas de investigación futuras basadas en las necesidades del momento en la Ingeniería del Software. El documento proporciona una visión general del campo: 1) la investigación que se ha llevado a cabo recientemente, 2) cuando esfuerzo se necesita concentrar, y 3) la importancia del diseño basado en escenarios.

La literatura revisada también enfatiza en la obtención y representación de las necesidades y de los problemas relacionados con la elicitación de los cambios en los requisitos (Pohl, 1994; Dubois y Pohl, 2003); en la negociación y la interpretación de los requisitos para desarrollar aplicaciones de software de calidad (Byrd, Cossick y Zmud, 1992; McKeen, Guimaraes y Wetherbe, 1994; Keil y Carmel de 1995; Markus y Mao, 2004). El enfoque de los escenarios abarca un proceso cíclico (Nguyen y Swatman, 2003) a partir de ideas vagas que se presentan de manera informal y a menudo incoherente, lo que lleva a un estado en el que existe un acuerdo común sobre de que puede servir como modelo para la elicitación de requisitos (Pohl, 1994).

Ventajas y desventajas de los escenarios en la elicitación de requisitos

En Ciencias Computacionales, un escenario es una narrativa de las posibles interacciones de los roles de los clientes, usuarios y sistemas (conocidos como partes interesadas o actores) en el desarrollo del ciclo de vida de un producto software o hardware. En la elicitación de requisitos su objetivo suele ser funcional y describe una manera de interpretar las necesidades de los actores en un marco de tiempo definido: una interacción, una transacción, una operación de negocios, un día u otro período, o toda la vida operativa del sistema. Además, su alcance podría ser un componente, un subsistema o un Sistema de Información completo (Alexander y Ljerka, 2009). Poseen la característica de estar escritos en un lenguaje sencillo y con mínimos detalles técnicos, de manera que las partes interesadas los puedan comprender y hacerse a una idea sobre en qué centrar los análisis de la elicitación (Carroll, 2000). En esta técnica, el escenario se materializa de diversas formas: historias, mundos alternativos, simulaciones, storyboard, secuencias, o estructuras (Alexander y Maiden, 2004).

Gomaa y Scott (1981) indican que un buen momento para desarrollar un escenario es después de la primera versión de la especificación de requisitos, porque el desarrollador tiene una sólida comprensión del problema y ha hecho el primer intento

de satisfacer las necesidades de los usuarios. Sin embargo, estos autores no hacen ninguna sugerencia sobre cómo crear ese conjunto preliminar de requisitos. Se puede inferir que se necesita otro proceso para modelarlos rápida y económicamente. La técnica de escenarios, como una manera informal, intuitiva y de bajo costo para generar la retroalimentación de las partes interesadas, no está atada a ninguna metodología en particular y podría ser utilizada como ellos proponen en su trabajo.

Crowley (1982) observa que a través de las historias somos capaces de visualizar eventos y relaciones, porque que constituyen una forma casi universal para organizar y difundir experiencias e información; son un medio importante para crear significado social y un sentido compartido de participación. En la elicitación de requisitos, las historias se pueden ver como un escenario en el que se presenta una descripción idealizada, pero detallada, de una instancia específica de una interacción usuario-sistema (Young y Barnard, 1987). Wexelblat (1987) ve los escenarios como una manera informal y natural de describir cómo se comportan los sistemas y en su investigación busca explotar esta propiedad para facilitar una mayor comunicación entre los equipos de elicitación. Afirma que es evidente que el esfuerzo requerido para comprender las necesidades del usuario a través de escenarios, reduce drásticamente el costo de la Ingeniería de Requisitos. Además, los escenarios no aseguran la viabilidad y pueden ser vistos como idealizaciones por los usuarios no-técnicos. En consecuencia, Wexelblat advierte que los escenarios deben ser bien escritos para evitar verlos como una caja negra que resuelve los problemas de alguna manera inexplicable.

Para Holbrook (1990), los escenarios carecen de la generalidad de los prototipos y son algo más abstractos. Se desarrollan para hacerle frente a situaciones específicas, y no se prestan a la experimentación en situaciones diversas, y requieren un mayor grado de interpretación, pero requieren mucho menos documentación que las técnicas tradicionales. En el trabajo de Kyng (1995b) se describe que la técnica de elicitación por escenarios es una herramienta ampliamente utilizada por los equipos en la elicitación de requisitos. En colaboración con las partes interesadas, desde dentro o fuera de la organización, se generan múltiples escenarios de uso y funcionalidad del sistema requerido. Pero, debido a una de sus marcadas deficiencias, el manejo de las incertidumbres de los requisitos, esta técnica es a menudo desechada en la elicitación.

Por otro lado, los ingenieros con experiencia aprovechan las tendencias, las señales débiles y en ocasiones los requisitos comodines, para convertir los escenarios en historias relevantes y desafiantes.

Karen Allenby y Tim Kelly (2001) afirman que los son secuencias de acciones utilizadas para ilustrar el comportamiento del sistema, es decir, representan instancias operativas de uso del sistema. Para ellos, los escenarios representan cada uno los usos intencionales del sistema, pero tal vez bajo diferentes circunstancias o con diferentes pre-condiciones. Sin embargo, su objetivo (o su post-condición) siempre es la misma. En ellos, las excepciones al camino principal son errores que surgen en la ejecución, ya sea a través de interacciones con los actores o por mal funcionamiento del sistema. Esto es habitual para especificar cómo manejará el sistema estas excepciones. La identificación de estos eventos excepcionales y la especificación de su mitigación, es similar identificar fallas en el análisis de riesgos.

Los escenarios tienen la ventaja de describir los requisitos de la interfaz de usuario de las interacciones del usuario, capturadas o imaginadas, a través de ejemplos concretos de cómo lleva a cabo cada tarea (Fliedl, Kop y Mayr, 2005). Esta forma es mucho más representativa, para que un usuario final valide los requisitos de interfaz, que los modelos que utilizan principalmente los ingenieros de software. Porque su expresión es significativamente menos comprensible para los usuarios finales, que a menudo tienen problemas para validar estos requisitos con esos modelos (Vanderdonckt, 2005). En consecuencia, los escenarios permiten validar con eficacia y eficiencia estos requisitos de interfaz y para alimentar al resto del ciclo de vida de desarrollo.

Los escenarios de un sistema particular describen el método de trabajo de las diferentes sesiones de interacción o situaciones del sistema (Sutcliffe, 1998; Pfleeger, 1998). En la elicitación de requisitos, los escenarios son útiles (Sutcliffe, 2003): 1) para analizar la flexibilidad de las diferentes sesiones del sistema para encontrar requisitos, y 2) las respuestas de los usuarios después de la interacción con los escenarios ofrece flexibilidad para encontrar los requisitos. Generalmente, los escenarios se utilizan después de la especificación de requisitos iniciales (Behrens, 2002; Kotonya y Somerville, 2006; do Prado et al., 2007). Los escenarios se estructuran luego de elicitar los requisitos iniciales y de conocer la idea básica acerca de las funciones que deben

proporcionar para el sistema. Estos escenarios desarrollados se utilizan para encontrar y preparar la especificación detallada de los requisitos (Gunda, 2008).

En la Tabla 1 se presenta un resumen de las ventajas y desventajas de la técnica de escenarios en la elicitación de requisitos, que se obtuvo del análisis a los trabajos seleccionados.

Tabla 1. Ventajas y desventajas de la técnica de escenarios en la elicitación de requisitos

Ventajas	Desventajas
<ul style="list-style-type: none"> ▪ Proporciona una idea básica de los procesos de razonamiento y del conocimiento implícito acerca del problema en los actores participantes. ▪ Desafía a las partes interesadas a pensar en términos creativos y de futuro, más allá de una aproximación temprana a la solución del problema. ▪ Cuestiona suposiciones acerca del futuro sistema y sobre los factores y las fuerzas que influyen en el problema. ▪ Se basa en los recursos internos de una empresa, así como de un amplio grupo de partes interesadas. ▪ Fomenta el trabajo en equipo, la comunicación, la planificación y la confianza en las diferentes partes interesadas. ▪ Complementa e ilumina la planificación estratégica estándar para encontrar una solución al problema. ▪ Les ayudan a las personas a reflejar el día a día del negocio. ▪ Permiten comprender y respetar las diferentes cosmovisiones culturales de las partes interesadas. ▪ Su visión del problema posibilita que el equipo pueda hablar de características futuras del sistema. ▪ Desafían el uso y a reflejar la imaginación. ▪ Tienen en cuenta una amplia gama de posibilidades y toma de decisiones en torno a qué requisitos validar o negar. ▪ Como técnica centrada en el usuario, ayuda a asegurar que se desarrolle el sistema correcto mediante la elicitación de los requisitos desde ese punto de vista. ▪ Son una poderosa técnica para elicitar y documentar los requisitos funcionales. ▪ Debido a que están escritos en lenguaje natural, son fáciles de entender y facilitan la comunicación con los clientes y usuarios. ▪ Puede ayudar a manejar la complejidad de los grandes proyectos, porque descompone el problema en sus funciones principales y especifica las aplicaciones desde la perspectiva de los usuarios. ▪ Apoyan indirectamente el concepto de mecanismos de la POO, una especie de patrón que capta cómo los objetos colaboran para proporcionar un comportamiento que satisface cada requisito. ▪ Constituyen una buena base para la verificación de los modelos de nivel superior y para la validación de los requisitos funcionales. ▪ Proporcionan un medio objetivo de seguimiento al proyecto. ▪ Forman la base sobre la que especifican los requisitos. ▪ Al aplicarla en combinación con prototipos, frecuente ayuda a obtener mejoras en el diseño. 	<ul style="list-style-type: none"> ▪ Amplia dependencia de los auto-reportes ▪ Su naturaleza funcional conduce naturalmente a la descomposición funcional del sistema en términos de casos concretos y abstractos, que están relacionados por extensiones y asociaciones de puntos de vista. ▪ Debido a que los usuarios y los ingenieros provienen de diferentes disciplinas, es necesario utilizar diferentes conceptos, terminologías, técnicas y notaciones. ▪ No es fácil poder determinar cuándo parar. ▪ Aunque son abstracciones funcionales, no es posible aplicarles el modelado general y común a todas las técnicas tradicionales útiles para el análisis y el diseño de abstracciones funcionales. ▪ Debido a que son creaciones con alto nivel de abstracción que se generan antes de identificar los objetos y clases, ignoran su encapsulación y operaciones. ▪ Un problema importante para modelarlos es la falta de formalidad en las definiciones de los términos y expresiones involucrados. ▪ Puede ser que se estructure a ciegas la arquitectura temprana del sistema. ▪ Generalmente, se definen en términos de la interacción entre uno o dos actores, aunque la elicitación se haga en grupo. ▪ Tiene limitaciones como base para el desarrollo incremental y el seguimiento del proyecto. ▪ Un desarrollo de escenarios realistas puede llevar mucho tiempo y no hay ninguna garantía de que los resultados se implementen en realidad. ▪ Los escenarios pueden (pero no deberían) limitar al equipo de trabajo a examinar solamente las combinaciones de intervenciones de rendimiento incluidas en ellos. ▪ Es más exhaustiva que cualquiera otra técnica, pero se requieren más recursos para aplicarla. ▪ Tiene cierto potencial para el descubrimiento de incógnitas, pero su éxito depende del tamaño de la muestra y la diversidad de requisitos. ▪ Depende de la comunicación en lenguaje natural, por lo que corre el riesgo de interpretaciones ambiguas.

- Permiten la toma de decisiones a través del examen a múltiples alternativas, en lugar de soluciones individuales.
- En lugar de revisar requisitos como alternativas discretas, pueden proporcionar contextos para la especificación de requisitos.
- Tienen la capacidad de proporcionar una ilustración visual de las diferentes alternativas en el contexto de la elicitación.
- Tienen la ventaja de describir los requisitos, capturados o imaginarios, a través de ejemplos concretos de cómo llevan a cabo los usuarios su tarea.
- Permiten validar con eficacia y eficiencia los requisitos de interfaz de usuario y alimentan el resto del ciclo de vida de desarrollo.
- Analizan la flexibilidad de las diferentes sesiones del sistema para encontrar requisitos no especificados.
- Las respuestas de los usuarios después de la interacción con los escenarios, ofrecen flexibilidad para encontrar requisitos funcionales.
- Proporciona flexibilidad para encontrar requisitos técnicos y no-técnicos.
- Es una técnica económica para elicitación de requisitos.
- Se basa en ejemplos aterrizados sobre una realidad concreta, lo que estimula la comprensión y genera preguntas.

ANÁLISIS DE RESULTADOS

Si bien para muchos autores e investigadores la técnica de escenarios no es de utilidad en la mayoría de sus proyectos, otros la aplican cotidianamente por los logros que alcanzan en la elicitación de requisitos. Quienes no la utilizan trabajan mayoritariamente con proyectos complejos y de alta exigencia, y consideran que con esa técnica no es posible determinar parámetros de precisión como errores estándar, intervalos de confianza de las estimaciones, metadatos, normalización, codificación, ponderación de la falta de respuesta, error en el reporte, diseño de la muestra y recuento de casos, y que cuando se logra alguno de ellos, es muy poca la utilidad para la especificación de requisitos.

Para los demás esta cuestión es diferente, porque trabajan con proyectos de tamaño medio o pequeño, y consideran que la técnica de escenarios para la elicitación de requisitos no sustituye totalmente la exposición completa y objetiva del equipo a la aplicación de otras técnicas, sino que, por el contrario, es un complemento efectivo. En la elicitación tradicional, la información de las partes interesadas se utiliza para modelar el problema, con una especificación razonada, y las técnicas le permiten al analista afirmar, dentro de cierto porcentaje de error, que la especificación satisface las necesidades de clientes y usuarios. En este punto, y de acuerdo con los resultados de esta investigación, los escenarios son un complemento a esas técnicas, porque permite una mejor comprensión de los requisitos técnicos del sistema, es decir, aquellos que no es posible encontrar en las entrevistas o las reuniones con los usuarios.

Se puede afirmar entonces, que la utilidad de la técnica de escenarios para la elicitación de requisitos es directamente proporcional al tipo y tamaño del proyecto en el que se desea aplicar. Para proyectos grandes y complicados, puede que no resulte de mucha utilidad, porque no es fácil representar mediante escenarios todos los caminos y rutas alternas que una solución podría ofrecer. Mientras que en proyectos menores podría ofrecer alternativas complementarias a las técnicas tradicionales. Según el tipo de proyecto, los escenarios le ayudan al equipo de Ingeniería de Requisitos a integrarse a comprender las necesidades de los usuarios. Por ejemplo, en un problema muy técnico, representar mediante escenarios las funcionalidades del sistema le permite al usuario lego la comprensión necesaria para aportar desde su experiencia. Porque, cuando se le describe en palabras podría no comprender el requisito, mientras que con un escenario se le presenta una imagen comprensible de la situación de análisis.

De acuerdo con la literatura analizada, las ventajas y desventajas de los escenarios se analizan desde la óptica de cada actor involucrado en la elicitación. La recomendación es que el líder del equipo de trabajo debe tener la experiencia necesaria para decidir en qué situaciones utilizarla y cuáles no. Porque se enfrenta a otro problema en la conformación del equipo, al hecho de que los integrantes provienen casi todos de la misma disciplina (aquella en la que se encuentra inmerso el problema a resolver), por lo que su interpretación está sesgada a ella. La habilidad de líder debe trascender el método para la elicitación y superar estas barreras con una selección de técnicas ajustadas a cada componente del sistema. Desde los requisitos funcionales a los no-funcionales, los escenarios pueden ayudarle, o pueden convertirse en su peor enemigo.

Entonces, ¿qué deben hacer estos equipos? Claramente, los escenarios ofrecen muchos beneficios importantes y son un arma poderosa que probablemente debería estar en el arsenal de los analistas de requisitos, los diseñadores y los probadores. Sin embargo, lamentablemente es una técnica más funcional que orientada a objetos, y puede comprometer significativamente los beneficios de esta tecnología, si se aplica a ciegas en el proceso de la elicitación. Afortunadamente, los riesgos asociados con el modelado de escenarios pueden ser mitigados a través del conocimiento, la formación, el diálogo y de evitar una sobrevaloración entusiasta.

Los escenarios deben ser solamente una de varias formas de elicitar las necesidades del usuario. El modelado de objetos, clases y de sus relaciones semánticas debe ser coherente con, pero no totalmente impulsado por, los escenarios. Los diseñadores deben tener cuidado y minimizar la dispersión de las características de un escenario y deben tener cuidado para evitar la creación de variantes parciales o clases redundantes, sobre todo en proyectos grandes que involucran múltiples construcciones y lanzamientos.

5. CONCLUSIONES

Las opiniones que discuten en esta investigación sobre la técnica de escenarios en la elicitación de requisitos, reconocen y valoran la utilidad y potencialidad de esta herramienta para la elicitación. Aunque no se encontraron principios de homogeneidad entre ellas. Desde finales de 1970 y principios de los 80, los desarrolladores de sistemas han reconocido el valor de los escenarios. Sus enfoques de aplicación en la historia han sido informales, intuitivos y de bajo costo, con el objetivo de generar retroalimentación de las partes interesadas. Los escenarios no estado atados a ningún método particular y pueden ser utilizados como lo proponga el líder del equipo de la elicitación. Actualmente, los escenarios son particularmente valiosos en el proceso de la elicitación de requisitos, donde son utilizados en un número de diferentes actividades. Esta aceptación se debe a que esta generación de ingenieros es más visual que textual, y sus discusiones giran en torno a un escenario dirigido por storyboard.

En la segunda mitad de la década de 1990, había una multitud de métodos para la utilización de escenarios en todo el proceso de la Ingeniería de Requisitos. En esa década, los investigadores más importantes en este campo colaboraron con propuestas integradoras para mejorar esta técnica, lo que generó un marco para organizar todos los métodos de escenarios en un cuerpo unificado y coherente de trabajo. Entonces, la investigación se hizo instrumental y se anunció el uso del enfoque de escenarios para la elicitación, pero basado desde el principio en un proceso de diseño del sistema. Las investigaciones analizadas en esta revisión de la literatura, muestra una aceptación más amplia de esta técnica, y una tendencia a elegirla de entre la gran variedad de técnicas y métodos para elicitar requisitos.

Al comienzo del nuevo milenio, los investigadores trataron de darle sentido a toda la investigación previa en requisitos y escenarios. Surgieron nuevas ideas que les mostraron a los investigadores diversas formas en las que podría aplicarse la narrativa a los enfoques basados en escenarios. Sin embargo, ninguno de los

trabajos más recientes describe un estudio de investigación rigurosa que demuestre cómo el proceso de escritura de guiones puede ser especialmente útil en la elicitación de requisitos. Esto se debe a que, a pesar de la importancia de esta técnica en el proceso de obtención de requisitos, la experiencia demuestra dificultades en el descubrimiento de los objetivos del sistema. Y, a pesar de la concreción de los escenarios para capturar algunos aspectos relevantes del problema, todavía plantean problemas de cobertura y de integridad. Una posible solución para mitigar estas dificultades sería separar el logro de objetivos de los escenarios, para luego combinarlos en la especificación de requisitos.

Otra cuestión que ha llevado a la utilización de los escenarios en la elicitación de requisitos, es la falta de madurez de muchas de las tecnologías documentadas desde hace tiempo. Los equipos de trabajo que utilizan dichas tecnologías requieren desarrollar (inventar) nuevas maneras para obtener los requisitos, mientras que los escenarios echan abajo los argumentos y razonamientos que impiden la comprensión plena de un requisito al añadir detalles o ejemplos específicos. Por eso es que para muchos investigadores, esta técnica encaja en el proceso de la elicitación de requisitos, para lo cual recogen ejemplos e historias de los interesados que generalizan en sus trabajos. La elicitación de requisitos se puede convertir en un consumo de recursos innecesario y en un trabajo que consume mucho tiempo. De ahí que en esta investigación se hallan encontrado diversas propuestas para realizarla con base en la técnica de escenarios, lo que, según ellos, haría sencillo que la especificación de requisitos concuerde con las necesidades del sistema. Para trabajo futuro se espera que los investigadores expandan sus experimentos y propuestas de utilización de los escenarios a los requisitos no-funcionales, porque actualmente esta técnica se encuentra muy alejado de su elicitación.

REFERENCIAS

- Alexander, I. y Ljerka, B. (2009). *Discovering requirements: How to specify products and services*. New York: Wiley.
- Alexander, I. y Maiden, N. (2004). *Scenarios, stories, use cases - Through the systems development life cycle*. Chichester: John Wiley & Sons.
- Allenby, K. y Kelly, T. (2001). Deriving safety requirements using scenarios. *Proceedings Fifth IEEE International Symposium on Requirements Engineering* (pp. 228-235). Toronto, Canada.
- Alter, S. y Ginzberg, M. (1978). Managing uncertainty in MIS implementation. *Sloan Management Review* 20(1), pp. 23-31.
- Andrea, H. y Zmud, R. (2002). A contingency approach to software project coordination. *Journal of Management Information Systems* 18(3), pp. 41-70.
- Antón, A., Potts, C. y Takahashi, K. (1994a). Inquiry-based scenario analysis of systems requirements. Technical report GIT-CC-94/14. Georgia Institute of Technology.
- Antón, A., Potts, C. y Takahashi, K. (1994b). Inquiry-based requirements analysis. *IEEE Software* 11(2), pp. 21-32.
- Asur, S. & Hufnagel, S. (1993). Taxonomy of rapid-prototyping methods and tools. *Proceedings Fourth International Workshop on Rapid System Prototyping* (pp. 42-56). June 28-30, Research Triangle Park, USA.
- Barki, H., Rivard, S. y Talbot, J. (2001). An integrative contingency model of software project risk management. *Journal of Management Information Systems* 17(4), pp. 37-69.
- Behrens, H. (2002). *Requirements Analysis Using Statecharts and Generated Scenarios*. Doctoral Symposium at IEEE Joint Conference on Requirements Engineering. Essen, Germany.
- Byrd, T., Cossick, K. y Zmud, R. (1992). A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly* 16(1), pp. 117-138.
- Carroll, J. (2000). *Making use: Scenario-based design of human-computer interactions*. Massachusetts: MIT Press.
- Carroll, J. et al. (1998). Requirements development in scenario-based design. *IEEE transactions on Software Engineering* 24(12), pp. 1156-1170.
- Carroll, J. y Go, K. (2004). The blind men and the elephant: views of scenario-based system design. *Interactions* 11(6), pp. 44-53.
- Chatzoglou, P. y Macaulay, L. (1996). Requirements capture and IS methodologies. *Information Systems Journal* 6(3), pp. 209-225.
- Chen, C. et al. (1994). Formal approach to scenario analysis found. *IEEE Software* 11(2), pp. 33-34.
- Cheng, B. y Atlee, J. (2007). Research directions in Requirements Engineering. *Proceedings Future of Software Engineering* (pp. 285-303). Minneapolis, USA.
- Crowley, D. (1982). *Understanding communication: The signifying web*. New York: Gordon and Breach Science Publishers.
- Davis, A. y Hickey, A. (2003). Requirements elicitation and elicitation technique selection: A model for two knowledge-intensive software development processes. *Proceedings thirty-sixth annual Hawaii international conference on systems sciences* (pp. 1-10). Big Island, USA.
- Davis, G. (1982). Strategies for information requirements determination. *IBM Systems Journal* 21(1), pp. 4-31.

- de Roeber, W. y Engelhardt, K. (1998). Data refinement: Model-oriented proof methods and their comparison. Cambridge: Cambridge University Press.
- do Prado, J. et al. (2007). Enhancing a requirements baseline with scenarios. *Requirements engineering* 2(4), pp. 184-198.
- Dubois, E. and Pohl, K. (2003). RE 02: A major step toward a mature requirements engineering community. *IEEE Software* 20(1), pp. 14-15.
- Duggan, E. y Thachenkary, C. (2004). Integrating nominal group technique and joint application development for improved systems requirements determination. *Information & Management* 41(4), pp. 399-411.
- Fliedl, G., Kop, C. y Mayr, H. (2005). From textual scenarios to a conceptual schema. *Data Knowledge Engineering* 55(1), pp. 20-37.
- Frolick, M. y Robichaux, B. (1995). EIS information requirements determination - Using a group support system to enhance the strategic business objectives method. *Decision Support Systems* 14(2), pp. 157-170.
- Gomaa, H. y Scott, D. (1981). Prototyping as a tool in the specification of user requirements. *Proceedings 5th international conference on Software Engineering* (pp. 333-342). San Diego, USA.
- Gottesdiener, E. (2008). Good practices for developing user requirements. *CrossTalk: The journal of defense Software Engineering*, March, pp. 13-17.
- Gunda, S. (2008). Requirements Engineering - Elicitation Techniques. Master's Thesis. Department of Technology, Mathematics and Computer Science, Software University West.
- Hickey, A. y Davis, A. (2004). A unified model of requirements elicitation. *Journal of Management Information Systems* 20(4), pp. 65-84.
- Holbrook, H. (1990). A scenario-based methodology for conducting requirements elicitation. *Software Engineering Notes* 15(1), pp. 96-104. I
- Holbrook, H. (1990). A scenario-based methodology for conducting requirements elicitation. *SIGSOFT Software Engineering Notes* 15(1), pp. 95-104.
- Hooper, J. y Hsia, P. (1982). Scenario-based prototyping for requirements identification. *Proceedings of the Workshop on Rapid Prototyping* (pp. 88-93). Columbia, USA.
- Hruschka, P. (1997). Detailing and deriving system requirements. *Proceedings International Conference and Workshop on Engineering of Computer-Based Systems* (pp. 25-32). March 24-28, Monterrey, USA.
- Iivari, J. (1992). The organizational fit of information systems. *Journal of Information Systems* 2(1), pp. 3-29.
- Jacobson, I. (1992). *Object-oriented software engineering: A use case driven approach*. USA: Addison Wesley.
- Jayaratra, N. (1994). *Understanding and evaluating methodologies*. London: McGraw Hill.
- Keil, M. y Carmel, E. (1995). Customer-developer links in software development. *Communications of the ACM* 38(5), pp. 33-44.
- Khedri, R. (2003). Requirements scenarios formalization technique: *N* versions towards one good version. *Electronic Notes in Theoretical Computer Science* 44(3), pp. 112-135.
- Kickert, W. (1983). Research note: Research models underlying situational dependency. *Organization Studies* 4(1), pp. 55-72.
- Kimmond, R. (1995). Survey into the acceptance of prototyping in software development. *Proceedings IEEE International Workshop on Rapid System Prototyping* (pp. 147-153). June 7-9, North Carolina, USA.

- Kotonya, G. y Somerville, I. (2006). Requirements engineering process and techniques. USA: Willey.
- Kulak, D. & Guiney, E. (2000). Use cases - Requirements in context. New York: Addison Wesley.
- Kyng, M. (1995a). Creating contexts for design. In Carroll, J. (Ed.), Scenario-based design: Envisioning work and technology in system development (pp. 85-108). New York: Willey.
- Kyng, M. (1995b). Making representations work. Communications of the ACM 38 (9), pp. 46-55.
- Lamb, R. y Kling, R. (2003). Reconceptualizing users as social actors in information systems research. MIS Quarterly 27(2), pp. 197-235.
- Leffingwell, D. & Widrig, D. (1999). Managing software requirements - A unified approach. USA: Addison Wesley.
- Lyytinen, K. (1988). Expectation failure concept and systems analysts' view of information system failures: Results of an exploratory study. Information & Management 14(1), pp. 45-56.
- Lyytinen, K., Mathiassen, L. y Ropponen, J. (1998). Attention shaping and software risk - A categorical analysis of four classical risk management approaches. Information Systems Research 9(3), pp. 233-255.
- Markus, L. y Mao, J. (2004). Participation in development and implementation - Updating an old, tired concept for today's contexts. Journal of the Association for Information Systems 5(11-12), pp. 514-544.
- Mathiassen, L. (2007). A contingency model for requirements development. Journal of the Association for Information Systems 8(11), Article 33.
- McFarlan, F. (1981). Portfolio approach to information systems. Harvard Business Review 59(5), pp. 142-150.
- McFarlan, F. (1982). Portfolio approach to information systems. Journal of Systems Management 33(1), pp. 12-19.
- McKeen, J., Guimaraes, T. y Wetherbe, J. (1994). The relationship between user participation and user satisfaction - An Investigation of 4 contingency factors. MIS Quarterly 18(4), pp. 427-451.
- Neill, C. y Laplante, P. (2003). Requirements Engineering: The state of the practice. IEEE Software 20(6), pp. 40-45.
- Nguyen, L. y Swatman, P. (2003). Managing the requirements engineering process. Requirements Engineering Journal 8(1), pp. 55-68.
- Peffer, K., Gengler, C. y Tuunanen, T. (2003). Extending critical success factors methodology to facilitate broadly participative information systems planning. Journal of Management Information Systems 20(1), pp. 51-85.
- Pfleeger, S. (1998). Software Engineering - Theory and practices. New York: Prentice Hall.
- Pohl, K. (1994). The three dimensions of requirements engineering - A framework and its applications. Information Systems 19(3), pp. 243-258.
- Pohl, K. (1997). Requirements Engineering - An overview, encyclopedia of computer science and technology (Vol. 36). New York: Marcel Dekker.
- Potts, C. (1995). Using schematic scenarios to understand user needs. Proceedings 1st conference on designing interactive systems: Processes, practices, methods, & techniques (pp. 247-256). Michigan, USA.
- Schneider, G. & Winters, J. (2001). Applying use cases - A practical guide. New York: Addison Wesley.
- Serna, M.E. (2012). Analysis and selection to requirements elicitation techniques. Proceedings 7th Computing Colombian Congress (pp. 1-7). Medellin, Colombia.

- Serna, M.E. y Serna, A.A. (2015). MoDeMaRE: Model to Development and Management Requirements Engineering. IEEE Transactions on Software Engineering. In press.
- Sommerville, I. & Sawyer, P. (1997). Requirements engineering - A good practice guide. USA: Wiley.
- Sutcliffe, A. (1998). Scenario-based requirements analysis. Requirements Engineering 3(1), pp. 48-65.
- Sutcliffe, A. (2003). Scenario-based Requirements Engineering. Proceedings 11th IEEE International Conference on Requirements Engineering (pp. 320-326). California, USA.
- Sutcliffe, A. (2003). Scenario-based Requirements Engineering. Proceedings 11th IEEE International Requirements Engineering Conference (pp. 320-329). Monterey Bay, USA.
- Vanderdonckt, J. (2005). A MDA-compliant environment for developing user interfaces of Information Systems. Lecture Notes in Computer Science 3520, pp. 16-31.
- Webster, J. y Watson, R. (2002). Analyzing the past to prepare for the future: Writing a literature review. MIS Quarterly 26(2), pp. xiii-xx.
- Weidenhaupt, K. et al. (1998). Scenarios in system development: Current practice. IEEE Software 15(2): pp. 34-45.
- Weill, P. y Olson, M. (1989). An assessment of the contingency theory of management information systems. Journal of Management Information Systems 6(1), pp. 59-85.
- Wexelblat, A. (1987). Report on scenario technology. MCC technical report STP-139-87. Microelectronics and computer Technology Corporation. Austin: USA.
- Young, R. y Barnard, P. (1987). The use of scenarios in human-computer interaction research - Turbocharging the tortoise of cumulative science. ACM SIGCHI Bulletin 17(SI), pp. 291-296.
- Zhang, P. y Li, N. (2005). The intellectual development of human-computer interaction research: A critical assessment of the MIS literature (1990-2002). Journal of the Association for Information Systems 6(1), pp. 227-292.
- Zhang, Z. (2007). Effective requirements development - A comparison of requirements elicitation techniques. In Berki, E. et al. (Eds.), Software quality management XV: Software quality in the knowledge society (pp. 225-240). London: British Computer Society.

APENDICE (Artículo científico)

Análisis a la utilidad de la técnica de escenarios en la elicitación de requisitos

Analysis of the usefulness of scenarios technique in requirements elicitation Edgar

Serna M., Ángela Tascon C., Francisco Domínguez H.

Resumen

Se presenta un análisis a las ventajas y desventajas de la técnica de escenarios para la elicitación de requisitos. Mediante una investigación dirigida por una revisión de la literatura, se describen los resultados obtenidos de las opiniones y experimentos divulgados en 63 trabajos relacionados. De acuerdo con esta revisión, la técnica de escenarios tiene ventajas importantes frente a otras técnicas de elicitación, aunque no es muy práctica en proyectos grandes y complicados. En problemas de envergadura media y pequeña, los escenarios les facilitan a los usuarios una comprensión más precisa de la situación que se desea solucionar, debido a que su representación es más visual que técnica. Entre las desventajas se cita frecuentemente su dependencia del lenguaje natural y que son difíciles de documentar debido a la cantidad de texto que se requiere.

Palabras clave: Ingeniería de Requisitos, Ingeniería del Software, ciclo de vida, técnicas de elicitación, requisitos funcionales, requisitos no-funcionales.

Abstract

It is presented an analysis of the advantages and disadvantages of the scenario technique for requirements elicitation. Through an investigation conducted by a review of the literature, it is describe the results obtained of the opinions and experiments reported in 63 related work. According to this review, the scenario technique has significant advantages over other elicitation techniques, although it is not very practical in large and complicated projects. In problems of medium and small size, the scenarios facilitate to users a more precise comprehension of the situation to be solved, because their representation is more visual than technical. Among the disadvantages it is frequently cited its dependence on natural language and are difficult to document due the amount of text that is required.

Keywords: Requirements Engineering, Software Engineering, life cycle, elicitation techniques, functional requirements, non-functional requirements.

INTRODUCCIÓN

La elicitación de requisitos es una etapa de la Ingeniería de Requisitos en la que se descubren las necesidades que los usuarios requieren satisfacer con un sistema (Serna y Serna, 2015). En el proceso se utilizan diversas técnicas, tales como entrevistas, encuestas, observación, grupos focales, talleres, cuestionarios, análisis de documentos y reuniones creativas, en las que es necesaria una buena comunicación entre clientes, usuarios, ingenieros y otros que tengan algún interés en el desarrollo del sistema. Una de los problemas en esta fase es que muchas de esas necesidades, empresariales o técnicas, no están documentadas, sino que residen en la mente de los interesados, en la retroalimentación que aún no se ha obtenido de los usuarios finales y en los estudios de diagramas de procesos y encuestas que aún no se han creado. De esta manera, los requisitos deben ser elicitados, o extraídos, y la metodología seleccionada para hacerlo debe ser lógica y meticulosa.

Llevar a cabo una elicitación efectiva es importante y no se puede tomar a la ligera, porque la información obtenida en esta etapa es la base para la especificación de requisitos, que a su vez es la base para las demás fases dl ciclo de vida del producto. Se ha demostrado que los errores

cometidos en la elicitación suelen ser las principales causas del fracaso o el abandono de los proyectos y que tienen un costo muy alto (Zhang, 2007; Serna, 2012). De ahí la importancia de la elicitación, aunque mayoría de las herramientas de requisitos no la atienden específicamente. Y puede ser aún más complicado, cuando en el desarrollo de muchas aplicaciones el usuario no está disponible para entrevistas o reuniones de grupos focales. Si bien muchos de los enfoques de desarrollo tradicionales son aplicables a la mayoría de proyectos, el análisis de requisitos plantea retos especiales cuando los usuarios son anónimos o no están físicamente disponibles para participar en la elicitación. Esta situación se complica por la diversidad potencial que puede existir entre usuarios y clientes. Entonces, ¿cómo capturar los requisitos de los usuarios, cuando sus aportes son voluntarios, no son de fácil acceso y no existen incentivos suficientes para que participe en la elicitación?

Desde la última década del siglo XX se inició el desarrollo de un método que diversos autores recomiendan para salvar estas dificultades: los escenarios. Este método consiste en utilizar personajes para definir características y atributos de los distintos tipos de usuarios. Se documentan los supuestos clave acerca de ellos, tales como el nivel de conocimientos informáticos y del negocio, las habilidades, las horas de trabajo, las expectativas, y otros similares. El uso de escenarios se ha convertido en una buena práctica para la elicitación de requisitos, al punto que algunos ya la consideran una técnica efectiva en la ausencia física de las partes interesadas (Gottesdiener, 2008). Los escenarios se consideran un medio práctico y útil en la elicitación de requisitos, y Sutcliffe (2003) afirma que son ejemplos de sesiones de interacción que consisten en descripciones de acciones secuenciales, que se relacionan con ejemplos de la vida real, en lugar de descripciones abstractas de las funciones del producto. Las relaciones entre los usuarios y los sistemas se dan fácilmente a través de los escenarios, como una declaración abstracta de lo que éstos requieren de ellos. Además, son útiles para añadir información al entorno de la descripción del requisito. En general, un escenario contiene información no solamente sobre el funcionamiento del sistema, sino también sobre su entorno. Un escenario puede contener la descripción de las acciones en los sistemas manuales, descripciones de los actores, los roles y también de la organización.

El interés de la investigación que se presenta en este artículo radica en la comprensión de la utilidad de la técnica de escenarios cuando se aplica a diferentes contextos y proyectos en la elicitación de requisitos. Para eso fue necesario analizar las ventajas y desventajas que se reportan en la literatura y conocer las experiencias que la comunidad ha logrado en su aplicación. Se presenta un análisis a la relación, la importancia, las ventajas y las desventajas de utilizar los escenarios en la elicitación de requisitos. Se hace una descripción de los trabajos relacionados, luego se detalla el estado del arte en relación con la elicitación de requisitos y los escenarios. Se describe la metodología de la investigación; se presenta un análisis de los beneficios y problemas de los escenarios en la elicitación, y finalmente se concluye el trabajo realizado.

TRABAJOS RELACIONADOS

John Carroll y sus colegas (1998) describen y analizan el proceso del desarrollo de requisitos basado en escenarios, a través de la consideración de un estudio de caso. Utilizaron un grupo de profesores y desarrolladores de sistemas para crear un laboratorio de física virtual. Su trabajo de diseño se centró en el desarrollo colaborativo de una serie de escenarios para describir las actividades actuales y futuras de una clase en el laboratorio virtual. Emplearon el análisis de reclamaciones para evaluar el intercambio implícito en estos escenarios, y codificaron las ventajas y desventajas específicas en la elicitación de los requisitos. En el desarrollo del estudio de caso, la naturaleza de los requisitos del proyecto evolucionó, y no

solamente les proporcionó más información, sino también más tipos de información. En su trabajo discuten la utilidad de los escenarios en la elicitación y gestión de requisitos, y concluyen que es una nueva técnica que proporciona la información necesaria para la Ingeniería de Requisitos. Aunque no se detienen en detalles acerca de la mayoría de problemas que detectaron, este trabajo es interesante para una época en la que los modelos de ciclo de vida del software todavía estaban apegados a los tradicionales de los años 60 y 70.

Utilizando la noción de simulación introducida por de Roever y Engelhardt (1999), Ridha Khedri (2003) define una noción correspondiente que utiliza como una prueba para detectar las discrepancias entre los escenarios formales y los informales en la elicitación de requisitos. La técnica que utiliza autor le permite 1) detectar discrepancias formales entre las versiones de los escenarios; 2) validar, para una situación dada, la formalización obtenida por un escenario formal contra otro informal (con errores de formalización); y 3) detectar discrepancias formales entre los escenarios formales de diferentes escenarios informales que comparten al menos un componente. En este trabajo se aprecia un esfuerzo amplio por intentar formalizar la técnica de los escenarios, como parte de la iniciativa de los métodos formales para la Ingeniería de Requisitos. Aunque logra un avance importante, la técnica de Khedri se queda corta en el análisis a las ventajas y desventajas de los escenarios, sean formales o informales, para elicitar requisitos.

Basándose en una revisión de la literatura acerca del desarrollo de requisitos y de software, Lars Mathiassen y sus colegas (2007) presentan un estudio a 116 artículos. Analizan los riesgos del desarrollo de requisitos mediante escenarios y relacionan las técnicas y heurísticas encontradas en la revisión. Luego de sintetizar las ventajas y desventajas y los puntos de vista de la literatura y proponen un modelo para el desarrollo de requisitos con relación a patrones de resolución mediante escenarios. En los resultados obtenidos por estos autores se percibe la falta de un análisis comparativo de la técnica de escenarios con otras técnicas para elicitar requisitos. Aunque ese tampoco es el objetivo del presente trabajo, el artículo de Mathiassen et al. es muy amplio y extenso, por lo que pudieron haber incluido ese análisis.

Por su parte, Zhang (2007) afirma que el proceso de la Ingeniería de Requisitos es un esfuerzo humano, porque son las personas quienes identifican y documentan las necesidades para construir un producto software. La cuestión es que son de diferentes orígenes y con diferentes objetivos organizacionales e individuales, posiciones sociales y personalidades; tienen diferentes maneras de entender y expresar el conocimiento y de comunicarse con los demás. Por lo que los procesos del desarrollo de requisitos, como la elicitación, varían ampliamente dependiendo de las personas involucradas. Con el fin de elicitar los requisitos desde esta diversidad de fuentes, se ha propuesto un amplio número de técnicas. Sin embargo, debido a la falta de comprensión y a la variabilidad de las situaciones en las que se desarrollan los requisitos, es difícil identificar la más apropiada. Este autor analiza varias técnicas, entre ellas a la de escenarios, y concluye que ninguna es suficiente para desarrollar un proceso de Ingeniería de Requisitos, por lo que esta falencia se convierte en un factor importante que causa el fracaso de muchos proyectos.

Por el contrario, para Ellen Gottesdiener (2008) los escenarios es una de las mejores técnicas para definir las necesidades de los usuarios, aunque sin duda es uno de los retos más difíciles en la construcción de sistemas. Cuando se trata de elicitar los requisitos del usuario, es imprescindible el uso de modelos para documentarlos y analizarlos, y los mejores resultados se obtienen a través de los escenarios, aunque los usuarios no estén presentes. En su artículo proporciona un modelo de hoja de ruta para los requisitos, con el objetivo de ayudarles a los

equipos de desarrollo a entender el uso eficaz de las técnicas de elicitación, y describe un conjunto de buenas prácticas para su potencialización.

METODOLOGÍA

Una revisión de la literatura debe ser completa y centrada en conceptos. Por lo tanto, dos de las cuestiones relevantes que se deben resolver son cómo identificar la literatura relevante y cómo estructurar el análisis de los hallazgos (Weill y Olson, 1989; Webster y Watson, 2002; Zhang y Li, 2005). Para resolver la primera cuestión, en el desarrollo de la investigación se consideraron las publicaciones relacionadas con el desarrollo de Sistemas de Información, la Ingeniería de Requisitos y las técnicas para elicitar requisitos. Estas corrientes de la literatura ofrecen perspectivas relevantes para cumplir con el objetivo respectivo, primero, desde un punto de vista especializado (Davis, 1982) y segundo desde un punto de vista más general (McFarlan, 1981, 1982). Webster y Watson (2002) sugieren identificar artículos relevantes en las principales revistas y luego revisar las citas que se hacen de ellos; y Weill y Olson (1989) sugieren aplicar la crítica estructurada para guiar la selección de artículos adicionales.

Se utilizó el servicio Web of Science con acceso a la literatura científica para identificar los trabajos y las investigaciones pertinentes. Se utilizaron palabras clave generales y se identificaron 500 artículos relevantes en los temas de búsqueda. Posteriormente, se seleccionaron los artículos publicados en las revistas mejor clasificadas. En este proceso se definieron 113 artículos, pero muchos resultaron tratar la temática de forma periférica o sin relevancia para el estudio. Por ello, se realizó un filtro manual con base en el criterio de que debían evaluar la técnica de escenarios para elicitar requisitos, y se eliminaron 23 trabajos. Luego se revisaron las citas que se hicieron de ellos. En este proceso se compilaron los artículos que tenían tres o más citas en las principales revistas, con lo que la muestra final quedó conformada por 63 trabajos.

Para resolver la segunda cuestión, cómo estructurar el análisis de los hallazgos, se aplicó la teoría de contingencia con el apoyo del modelo unificado de elicitación de requisitos (Hickey y Davis, 2004), que sugiere utilizar factores contextuales como base para adoptar una técnicas de requisitos. Para superar las limitaciones de la teoría de contingencia clásica, se adoptó la gestión del riesgo como una forma específica de la teoría que se ha aplicado con éxito en la disciplina de Sistemas de Información (Lyytinen, 1988; Lyytinen, Mathiassen y Ropponen, 1998). Los modelos de contingencia y los enfoques de gestión de riesgos se basan en visiones particulares sobre cómo responder efectivamente en contextos específicos. Es por ello que se analiza la literatura identificada para entender cómo aplicar patrones de técnicas de elicitación de requisitos en arquetipos de perfiles de riesgo. El objetivo era determinar, mediante el riesgo y la contingencia, las ventajas y desventajas de las técnicas de elicitación presentadas en la literatura, especialmente sobre la técnica de escenarios.

RESULTADOS

Estado del arte

A mediados del siglo XX, los requisitos para las primeras aplicaciones software eran fáciles de identificar, porque la mayoría eran desarrolladas por los científicos para apoyar sus propias necesidades y propósitos. Sin embargo, a medida que las empresas comenzaron a desarrollar software para los usuarios, pronto se hizo necesario reunir, explicar y entender las necesidades de los usuarios, y que participaran en el proceso de desarrollo del producto (McKeen, Guimaraes y Wetherbe, 1994; Lamb y Kling, 2003; Markus y Mao, 2004). Esto ha dado lugar al surgimiento de una considerable variedad de técnicas para apoyar la elicitación de esas

necesidades (Davis, 1982; Byrd, Cossick y Zmud, 1992; Keil y Carmel, 1995; Chatzoglou y Macaulay, 1996). Algunos argumentan que el flujo constante de técnicas se ha convertido en una jungla metodológica (Jayaratna, 1994).

Buscando una solución, los investigadores han respondido proponiendo modelos de contingencia para la adopción de estas técnicas a contextos específicos. Algunos ofrecen categorías de factores contextuales, portafolios de técnicas y enfoques para adaptarlas a cada contexto (Kickert, 1983; Iivari, 1992); otros integran conocimiento dentro de determinadas disciplinas (Barki, Rivard y Talbot, 2001; Andrea y Zmud, 2002; Hickey y Davis, 2004) y a menudo se basan en modelos de gestión de riesgos (Lyytinen, Mathiassen y Ropponen, 1998). Unos de los primeros intentos por integrar el conocimiento disponible para elicitar requisitos, Alter y Ginzberg (1978) propusieron una técnica de contingencia para el soporte de decisiones; McFarlan (1981, 1982) sugirió el desarrollo de aplicaciones software a través de la integración interna entre los desarrolladores y externa entre desarrolladores y usuarios; y Davis (1982) se centró en el desarrollo de las necesidades de los usuarios integrando el conocimiento disponible mediante una técnica contingente para reducir su incertidumbre.

Por desgracia, y a pesar que la práctica y la investigación han cambiado considerablemente en los últimos años (Neill y Laplante, 2003), no ha habido intentos recientes para integrar los nuevos conocimientos disponibles acerca de la elicitación de requisitos (Hickey y Davis, 2004). Desde hace algún tiempo, la técnica de escenarios ha penetrado con énfasis en las aplicaciones entre organizaciones, y las nuevas demandas han introducido nuevas técnicas que han cambiado el perfil de la Ingeniería de Requisitos. Debido a la complejidad de los problemas actuales, los equipos de requisitos a menudo se enfrentan al reto de establecer una interacción efectiva con usuarios que están por fuera del alcance organizacional (Frolick y Robichaux, 1995; Peffers, Gengler y Tuunanen, 2003; Duggan y Thachenkary, 2004), un reto que se incrementa cuando los usuarios no saben cómo describir sus necesidades (Walz et al., 1993).

En cuanto al enfoque del modelo de escenarios, Hooper y Hsiao (1982) fueron los primeros en proponer su uso en la Ingeniería de Requisitos. Desarrollaron una metodología para identificar requisitos mediante escenarios, que sirven como un prototipo inicial rápido del sistema previsto. Chen et al. (1994) volvieron a examinar esta idea en su trabajo, y exploraron de manera sistemática cómo usar escenarios en el análisis de requisitos. Utilizaron una base matemática formal y generaron escenarios precisos, con capacidad para el cambio, y manteniendo a los usuarios involucrados en el proceso. Kyng (1995a) jugó un papel decisivo en la elaboración de la relación entre los escenarios y los prototipos. Su investigación en diseño cooperativo enfatizó el uso de herramientas de baja tecnología, como escenarios para apoyar el diseño. En su propuesta, los escenarios se desarrollan para los usuarios finales como un prototipo que contribuye al diseño en curso.

La primera referencia a la utilización de escenarios específicamente para elicitación de requisitos la presentó Holbrook (1990), quien describe una metodología que utiliza principios de la interacción entre los usuarios y diseñadores para desarrollar rápidamente un conjunto de requisitos iniciales usando escenarios. Su investigación proporciona una visión general de cómo utilizar los escenarios para refinar los objetivos y establecer los requisitos. Antón, Potts y Takahashi (1994a) proponen un modelo en el que describen cómo representar escenarios como ejecuciones de planes dirigidos a objetivos, proporcionando un puente entre los requisitos en la fase de planificación con los de la fase de análisis. Asimismo, exploraron aún más el modelo como una estructura formal y propusieron actividades como: elicitación, documentación y refinamiento (Antón, Potts y Takahashi, 1994b). Posteriormente, Potts

(1995) hizo una conexión directa entre las historias narrativas y el modelado de escenarios, y propuso una forma de aplicar ideas narrativas a las necesidades del usuario. Propuso directrices para definir los escenarios más destacados y cómo podrían ser utilizados en un sistema interactivo. Además, describe un método para escribir y utilizar escenarios para analizar las necesidades de los usuarios y del sistema.

Para finales del siglo XX había una multitud de métodos para utilizar escenarios en todo el proceso de la Ingeniería de Requisitos. En un enciclopédico trabajo, Pohl (1997) propuso un marco global para la caracterización del proceso de requisitos: los cuatro mundos del desarrollo, el asunto, el sistema y el uso; y tres dimensiones que incluyen acuerdo, representación y especificación. Los investigadores más importantes en el tema de requisitos colaboraron en el desarrollo de un marco para organizar todos los métodos de escenarios de entonces en un cuerpo unificado y coherente de trabajo. Achour et al. (1996) exploraron las cuestiones subyacentes a los enfoques basados en escenarios para la elicitación de requisitos, y propone un marco para clasificarlos a partir de puntos de vista: 1) del formulario, del contenido, del propósito y del ciclo de vida; 2) un conjunto de facetas asociado que caracteriza y clasifica un escenario; y 3) estas facetas son medidas mediante un conjunto de atributos relevantes. En este mismo sentido, la propuesta de Alexander y Maiden (2004) crea un marco para clasificar los enfoques de escenarios y para comunicar la investigación y el desarrollo del trabajo acerca de los sistemas basados en escenarios.

Davis y Hickey (2003) introdujeron un nuevo modelo unificado de elicitación de requisitos que mostraba el papel crítico de la selección del conocimiento y de la técnica. Alexander y Maiden (2004) trabajaron sobre la idea de que la complejidad de los sistemas se puede manejar si las necesidades de diseño se definen temprana y cuidadosamente. Propusieron a los escenarios como de las técnicas más poderosas para descubrir y comunicar los requisitos. Carroll and Go (2004) sugirieron unificar todos los métodos de planificación de escenarios en una jerarquía con base en las comunidades que emplean este método de diseño: la planificación estratégica, la Ingeniería de Requisitos, el diseño de interfaz humano-computador y el diseño orientado a objetos. Cheng y Atlee (2007) revisaron la investigación en Ingeniería de Requisitos durante la última década del siglo XX y se concentraron en la identificación de líneas de investigación futuras basadas en las necesidades del momento en la Ingeniería del Software. El documento proporciona una visión general del campo: 1) la investigación que se ha llevado a cabo recientemente, 2) cuando esfuerzo se necesita concentrar, y 3) la importancia del diseño basado en escenarios.

La literatura revisada también enfatiza en la obtención y representación de las necesidades y de los problemas relacionados con la elicitación de los cambios en los requisitos (Pohl, 1994; Dubois y Pohl, 2003); en la negociación y la interpretación de los requisitos para desarrollar aplicaciones de software de calidad (Byrd, Cossick y Zmud, 1992; McKeen, Guimaraes y Wetherbe, 1994; Keil y Carmel de 1995; Markus y Mao, 2004). El enfoque de los escenarios abarca un proceso cíclico (Nguyen y Swatman, 2003) a partir de ideas vagas que se presentan de manera informal y a menudo incoherente, lo que lleva a un estado en el que existe un acuerdo común sobre de que puede servir como modelo para la elicitación de requisitos (Pohl, 1994).

Ventajas y desventajas de los escenarios en la elicitación de requisitos

En Ciencias Computacionales, un escenario es una narrativa de las posibles interacciones de los roles de los clientes, usuarios y sistemas (conocidos como partes interesadas o actores) en el desarrollo del ciclo de vida de un producto software o hardware. En la elicitación de requisitos su objetivo suele ser funcional y describe una manera de interpretar las necesidades de los

actores en un marco de tiempo definido: una interacción, una transacción, una operación de negocios, un día u otro período, o toda la vida operativa del sistema. Además, su alcance podría ser un componente, un subsistema o un Sistema de Información completo (Alexander y Ljerka, 2009). Poseen la característica de estar escritos en un lenguaje sencillo y con mínimos detalles técnicos, de manera que las partes interesadas los puedan comprender y hacerse a una idea sobre en qué centrar los análisis de la elicitación (Carroll, 2000). En esta técnica, el escenario se materializa de diversas formas: historias, mundos alternativos, simulaciones, storyboard, secuencias, o estructuras (Alexander y Maiden, 2004).

Gomaa y Scott (1981) indican que un buen momento para desarrollar un escenario es después de la primera versión de la especificación de requisitos, porque el desarrollador tiene una sólida comprensión del problema y ha hecho el primer intento de satisfacer las necesidades de los usuarios. Sin embargo, estos autores no hacen ninguna sugerencia sobre cómo crear ese conjunto preliminar de requisitos. Se puede inferir que se necesita otro proceso para modelarlos rápida y económicamente. La técnica de escenarios, como una manera informal, intuitiva y de bajo costo para generar la retroalimentación de las partes interesadas, no está atada a ninguna metodología en particular y podría ser utilizada como ellos proponen en su trabajo.

Crowley (1982) observa que a través de las historias somos capaces de visualizar eventos y relaciones, porque que constituyen una forma casi universal para organizar y difundir experiencias e información; son un medio importante para crear significado social y un sentido compartido de participación. En la elicitación de requisitos, las historias se pueden ver como un escenario en el que se presenta una descripción idealizada, pero detallada, de una instancia específica de una interacción usuario-sistema (Young y Barnard, 1987). Wexelblat (1987) ve los escenarios como una manera informal y natural de describir cómo se comportan los sistemas y en su investigación busca explotar esta propiedad para facilitar una mayor comunicación entre los equipos de elicitación. Afirma que es evidente que el esfuerzo requerido para comprender las necesidades del usuario a través de escenarios, reduce drásticamente el costo de la Ingeniería de Requisitos. Además, los escenarios no aseguran la viabilidad y pueden ser vistos como idealizaciones por los usuarios no-técnicos. En consecuencia, Wexelblat advierte que los escenarios deben ser bien escritos para evitar verlos como una caja negra que resuelve los problemas de alguna manera inexplicable.

Para Holbrook (1990), los escenarios carecen de la generalidad de los prototipos y son algo más abstractos. Se desarrollan para hacerle frente a situaciones específicas, y no se prestan a la experimentación en situaciones diversas, y requieren un mayor grado de interpretación, pero requieren mucho menos documentación que las técnicas tradicionales. En el trabajo de Kyng (1995b) se describe que la técnica de elicitación por escenarios es una herramienta ampliamente utilizada por los equipos en la elicitación de requisitos. En colaboración con las partes interesadas, desde dentro o fuera de la organización, se generan múltiples escenarios de uso y funcionalidad del sistema requerido. Pero, debido a una de sus marcadas deficiencias, el manejo de las incertidumbres de los requisitos, esta técnica es a menudo desechada en la elicitación. Por otro lado, los ingenieros con experiencia aprovechan las tendencias, las señales débiles y en ocasiones los requisitos comodines, para convertir los escenarios en historias relevantes y desafiantes.

Karen Allenby y Tim Kelly (2001) afirman que los son secuencias de acciones utilizadas para ilustrar el comportamiento del sistema, es decir, representan instancias operativas de uso del sistema. Para ellos, los escenarios representan cada uno los usos intencionales del sistema, pero

tal vez bajo diferentes circunstancias o con diferentes pre-condiciones. Sin embargo, su objetivo (o su post-condición) siempre es la misma. En ellos, las excepciones al camino principal son errores que surgen en la ejecución, ya sea a través de interacciones con los actores o por mal funcionamiento del sistema. Esto es habitual para especificar cómo manejará el sistema estas excepciones. La identificación de estos eventos excepcionales y la especificación de su mitigación, es similar identificar fallas en el análisis de riesgos.

Los escenarios tienen la ventaja de describir los requisitos de la interfaz de usuario de las interacciones del usuario, capturadas o imaginadas, a través de ejemplos concretos de cómo lleva a cabo cada tarea (Fliedl, Kop y Mayr, 2005). Esta forma es mucho más representativa, para que un usuario final valide los requisitos de interfaz, que los modelos que utilizan principalmente los ingenieros de software. Porque su expresión es significativamente menos comprensible para los usuarios finales, que a menudo tienen problemas para validar estos requisitos con esos modelos (Vanderdonckt, 2005). En consecuencia, los escenarios permiten validar con eficacia y eficiencia estos requisitos de interfaz y para alimentar al resto del ciclo de vida de desarrollo.

Los escenarios de un sistema particular describen el método de trabajo de las diferentes sesiones de interacción o situaciones del sistema (Sutcliffe, 1998; Pfleeger, 1998). En la elicitación de requisitos, los escenarios son útiles (Sutcliffe, 2003): 1) para analizar la flexibilidad de las diferentes sesiones del sistema para encontrar requisitos, y 2) las respuestas de los usuarios después de la interacción con los escenarios ofrece flexibilidad para encontrar los requisitos. Generalmente, los escenarios se utilizan después de la especificación de requisitos iniciales (Behrens, 2002; Kotonya y Somerville, 2006; do Prado et al., 2007). Los escenarios se estructuran luego de elicitar los requisitos iniciales y de conocer la idea básica acerca de las funciones que deben proporcionar para el sistema. Estos escenarios desarrollados se utilizan para encontrar y preparar la especificación detallada de los requisitos (Gunda, 2008).

En la Tabla 1 se presenta un resumen de las ventajas y desventajas de la técnica de escenarios en la elicitación de requisitos, que se obtuvo del análisis a los trabajos seleccionados.

Tabla 1. Ventajas y desventajas de la técnica de escenarios en la elicitación de requisitos

Ventajas	Desventajas
<ul style="list-style-type: none"> ▪ Proporciona una idea básica de los procesos de razonamiento y del conocimiento implícito acerca del problema en los actores participantes. ▪ Desafía a las partes interesadas a pensar en términos creativos y de futuro, más allá de una aproximación temprana a la solución del problema. ▪ Cuestiona suposiciones acerca del futuro sistema y sobre los factores y las fuerzas que influyen en el problema. ▪ Se basa en los recursos internos de una empresa, así como de un amplio grupo de partes interesadas. ▪ Fomenta el trabajo en equipo, la comunicación, la planificación y la confianza en las diferentes partes interesadas. ▪ Complementa e ilumina la planificación estratégica estándar para encontrar una solución al problema. ▪ Les ayudan a las personas a reflejar el día a día del negocio. ▪ Permiten comprender y respetar las diferentes cosmovisiones culturales de las partes interesadas. ▪ Su visión del problema posibilita que el equipo pueda hablar de características futuras del sistema. 	<ul style="list-style-type: none"> ▪ Amplia dependencia de los auto-reportes ▪ Su naturaleza funcional conduce naturalmente a la descomposición funcional del sistema en términos de casos concretos y abstractos, que están relacionados por extensiones y asociaciones de puntos de vista. ▪ Debido a que los usuarios y los ingenieros provienen de diferentes disciplinas, es necesario utilizar diferentes conceptos, terminologías, técnicas y notaciones. ▪ No es fácil poder determinar cuándo parar. ▪ Aunque son abstracciones funcionales, no es posible aplicarles el modelado general y común a todas las técnicas tradicionales útiles para el análisis y el diseño de abstracciones funcionales. ▪ Debido a que son creaciones con alto nivel de abstracción que se generan antes de identificar los objetos y clases, ignoran su encapsulación y operaciones. ▪ Un problema importante para modelarlos es la falta de formalidad en las definiciones de los términos y expresiones involucrados.

- Desafían el uso y a reflejar la imaginación.
- Tienen en cuenta una amplia gama de posibilidades y toma de decisiones en torno a qué requisitos validar o negar.
- Como técnica centrada en el usuario, ayuda a asegurar que se desarrolle el sistema correcto mediante la elicitación de los requisitos desde ese punto de vista.
- Son una poderosa técnica para elicitar y documentar los requisitos funcionales.
- Debido a que están escritos en lenguaje natural, son fáciles de entender y facilitan la comunicación con los clientes y usuarios.
- Puede ayudar a manejar la complejidad de los grandes proyectos, porque descompone el problema en sus funciones principales y especifica las aplicaciones desde la perspectiva de los usuarios.
- Apoyan indirectamente el concepto de mecanismos de la POO, una especie de patrón que capta cómo los objetos colaboran para proporcionar un comportamiento que satisface cada requisito.
- Constituyen una buena base para la verificación de los modelos de nivel superior y para la validación de los requisitos funcionales.
- Proporcionan un medio objetivo de seguimiento al proyecto.
- Forman la base sobre la que especifican los requisitos.
- Al aplicarla en combinación con prototipos, frecuente ayuda a obtener mejoras en el diseño.
- Permiten la toma de decisiones a través del examen a múltiples alternativas, en lugar de soluciones individuales.
- En lugar de revisar requisitos como alternativas discretas, pueden proporcionar contextos para la especificación de requisitos.
- Tienen la capacidad de proporcionar una ilustración visual de las diferentes alternativas en el contexto de la elicitación.
- Tienen la ventaja de describir los requisitos, capturados o imaginarios, a través de ejemplos concretos de cómo llevan a cabo los usuarios su tarea.
- Permiten validar con eficacia y eficiencia los requisitos de interfaz de usuario y alimentan el resto del ciclo de vida de desarrollo.
- Analizan la flexibilidad de las diferentes sesiones del sistema para encontrar requisitos no especificados.
- Las respuestas de los usuarios después de la interacción con los escenarios, ofrecen flexibilidad para encontrar requisitos funcionales.
- Proporciona flexibilidad para encontrar requisitos técnicos y no-técnicos.
- Es una técnica económica para elicitar requisitos.
- Se basa en ejemplos aterrizados sobre una realidad concreta, lo que estimula la comprensión y genera preguntas.

- Puede ser que se estructure a ciegas la arquitectura temprana del sistema.
- Generalmente, se definen en términos de la interacción entre uno o dos actores, aunque la elicitación se haga en grupo.
- Tiene limitaciones como base para el desarrollo incremental y el seguimiento del proyecto.
- Un desarrollo de escenarios realistas puede llevar mucho tiempo y no hay ninguna garantía de que los resultados se implementen en realidad.
- Los escenarios pueden (pero no deberían) limitar al equipo de trabajo a examinar solamente las combinaciones de intervenciones de rendimiento incluidas en ellos.
- Es más exhaustiva que cualquiera otra técnica, pero se requieren más recursos para aplicarla.
- Tiene cierto potencial para el descubrimiento de incógnitas, pero su éxito depende del tamaño de la muestra y la diversidad de requisitos.
- Depende de la comunicación en lenguaje natural, por lo que corre el riesgo de interpretaciones ambiguas.

ANÁLISIS DE RESULTADOS

Si bien para muchos autores e investigadores la técnica de escenarios no es de utilidad en la mayoría de sus proyectos, otros la aplican cotidianamente por los logros que alcanzan en la elicitación de requisitos. Quienes no la utilizan trabajan mayoritariamente con proyectos complejos y de alta exigencia, y consideran que con esa técnica no es posible determinar parámetros de precisión como errores estándar, intervalos de confianza de las estimaciones,

metadatos, normalización, codificación, ponderación de la falta de respuesta, error en el reporte, diseño de la muestra y recuento de casos, y que cuando se logra alguno de ellos, es muy poca la utilidad para la especificación de requisitos.

Para los demás esta cuestión es diferente, porque trabajan con proyectos de tamaño medio o pequeño, y consideran que la técnica de escenarios para la elicitación de requisitos no sustituye totalmente la exposición completa y objetiva del equipo a la aplicación de otras técnicas, sino que, por el contrario, es un complemento efectivo. En la elicitación tradicional, la información de las partes interesadas se utiliza para modelar el problema, con una especificación razonada, y las técnicas le permiten al analista afirmar, dentro de cierto porcentaje de error, que la especificación satisface las necesidades de clientes y usuarios. En este punto, y de acuerdo con los resultados de esta investigación, los escenarios son un complemento a esas técnicas, porque permite una mejor comprensión de los requisitos técnicos del sistema, es decir, aquellos que no es posible encontrar en las entrevistas o las reuniones con los usuarios.

Se puede afirmar entonces, que la utilidad de la técnica de escenarios para la elicitación de requisitos es directamente proporcional al tipo y tamaño del proyecto en el que se desea aplicar. Para proyectos grandes y complicados, puede que no resulte de mucha utilidad, porque no es fácil representar mediante escenarios todos los caminos y rutas alternas que una solución podría ofrecer. Mientras que en proyectos menores podría ofrecer alternativas complementarias a las técnicas tradicionales. Según el tipo de proyecto, los escenarios le ayudan al equipo de Ingeniería de Requisitos a integrarse a comprender las necesidades de los usuarios. Por ejemplo, en un problema muy técnico, representar mediante escenarios las funcionalidades del sistema le permite al usuario lego la comprensión necesaria para aportar desde su experiencia. Porque, cuando se le describe en palabras podría no comprender el requisito, mientras que con un escenario se le presenta una imagen comprensible de la situación de análisis.

De acuerdo con la literatura analizada, las ventajas y desventajas de los escenarios se analizan desde la óptica de cada actor involucrado en la elicitación. La recomendación es que el líder del equipo de trabajo debe tener la experiencia necesaria para decidir en qué situaciones utilizarla y cuáles no. Porque se enfrenta a otro problema en la conformación del equipo, al hecho de que los integrantes provienen casi todos de la misma disciplina (aquella en la que se encuentra inmerso el problema a resolver), por lo que su interpretación está sesgada a ella. La habilidad de líder debe trascender el método para la elicitación y superar estas barreras con una selección de técnicas ajustadas a cada componente del sistema. Desde los requisitos funcionales a los no-funcionales, los escenarios pueden ayudarle, o pueden convertirse en su peor enemigo.

Entonces, ¿qué deben hacer estos equipos? Claramente, los escenarios ofrecen muchos beneficios importantes y son un arma poderosa que probablemente debería estar en el arsenal de los analistas de requisitos, los diseñadores y los probadores. Sin embargo, lamentablemente es una técnica más funcional que orientada a objetos, y puede comprometer significativamente los beneficios de esta tecnología, si se aplica a ciegas en el proceso de la elicitación. Afortunadamente, los riesgos asociados con el modelado de escenarios pueden ser mitigados a través del conocimiento, la formación, el diálogo y de evitar una sobrevaloración entusiasta.

Los escenarios deben ser solamente una de varias formas de elicitar las necesidades del usuario. El modelado de objetos, clases y de sus relaciones semánticas debe ser coherente con, pero no totalmente impulsado por, los escenarios. Los diseñadores deben tener cuidado y minimizar la dispersión de las características de un escenario y deben tener cuidado para evitar

la creación de variantes parciales o clases redundantes, sobre todo en proyectos grandes que involucran múltiples construcciones y lanzamientos.

CONCLUSIONES

Las opiniones que discuten en esta investigación sobre la técnica de escenarios en la elicitación de requisitos, reconocen y valoran la utilidad y potencialidad de esta herramienta para la elicitación. Aunque no se encontraron principios de homogeneidad entre ellas. Desde finales de 1970 y principios de los 80, los desarrolladores de sistemas han reconocido el valor de los escenarios. Sus enfoques de aplicación en la historia han sido informales, intuitivos y de bajo costo, con el objetivo de generar retroalimentación de las partes interesadas. Los escenarios no estado atados a ningún método particular y pueden ser utilizados como lo proponga el líder del equipo de la elicitación. Actualmente, los escenarios son particularmente valiosos en el proceso de la elicitación de requisitos, donde son utilizados en un número de diferentes actividades. Esta aceptación se debe a que esta generación de ingenieros es más visual que textual, y sus discusiones giran en torno a un escenario dirigido por *storyboard*.

En la segunda mitad de la década de 1990, había una multitud de métodos para la utilización de escenarios en todo el proceso de la Ingeniería de Requisitos. En esa década, los investigadores más importantes en este campo colaboraron con propuestas integradoras para mejorar esta técnica, lo que generó un marco para organizar todos los métodos de escenarios en un cuerpo unificado y coherente de trabajo. Entonces, la investigación se hizo instrumental y se anunció el uso del enfoque de escenarios para la elicitación, pero basado desde el principio en un proceso de diseño del sistema. Las investigaciones analizadas en esta revisión de la literatura, muestra una aceptación más amplia de esta técnica, y una tendencia a elegirla de entre la gran variedad de técnicas y métodos para elicitación de requisitos.

Al comienzo del nuevo milenio, los investigadores trataron de darle sentido a toda la investigación previa en requisitos y escenarios. Surgieron nuevas ideas que les mostraron a los investigadores diversas formas en las que podría aplicarse la narrativa a los enfoques basados en escenarios. Sin embargo, ninguno de los trabajos más recientes describe un estudio de investigación riguroso que demuestre cómo el proceso de escritura de guiones puede ser especialmente útil en la elicitación de requisitos. Esto se debe a que, a pesar de la importancia de esta técnica en el proceso de obtención de requisitos, la experiencia demuestra dificultades en el descubrimiento de los objetivos del sistema. Y, a pesar de la concreción de los escenarios para capturar algunos aspectos relevantes del problema, todavía plantean problemas de cobertura y de integridad. Una posible solución para mitigar estas dificultades sería separar el logro de objetivos de los escenarios, para luego combinarlos en la especificación de requisitos.

Otra cuestión que ha llevado a la utilización de los escenarios en la elicitación de requisitos, es la falta de madurez de muchas de las tecnologías documentadas desde hace tiempo. Los equipos de trabajo que utilizan dichas tecnologías requieren desarrollar (inventar) nuevas maneras para obtener los requisitos, mientras que los escenarios echan abajo los argumentos y razonamientos que impiden la comprensión plena de un requisito al añadir detalles o ejemplos específicos. Por eso es que para muchos investigadores, esta técnica encaja en el proceso de la elicitación de requisitos, para lo cual recogen ejemplos e historias de los interesados que generalizan en sus trabajos. La elicitación de requisitos se puede convertir en un consumo de recursos innecesario y en un trabajo que consume mucho tiempo. De ahí que en esta investigación se hallan encontrado diversas propuestas para realizarla con base en la técnica de escenarios, lo que, según ellos, haría sencillo que la especificación de requisitos concuerde con las necesidades del sistema. Para trabajo futuro se espera que los investigadores expandan

sus experimentos y propuestas de utilización de los escenarios a los requisitos no-funcionales, porque actualmente esta técnica se encuentra muy alejado de su elicitación.

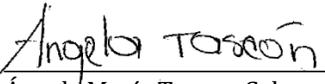
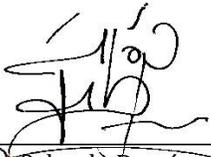
REFERENCIAS

- Alexander, I. y Ljerka, B. (2009). *Discovering requirements: How to specify products and services*. New York: Wiley.
- Alexander, I. y Maiden, N. (2004). *Scenarios, stories, use cases - Through the systems development life cycle*. Chichester: John Wiley & Sons.
- Allenby, K. y Kelly, T. (2001). Deriving safety requirements using scenarios. *Proceedings Fifth IEEE International Symposium on Requirements Engineering* (pp. 228-235). Toronto, Canada.
- Alter, S. y Ginzberg, M. (1978). Managing uncertainty in MIS implementation. *Sloan Management Review* 20(1), pp. 23-31.
- Andrea, H. y Zmud, R. (2002). A contingency approach to software project coordination. *Journal of Management Information Systems* 18(3), pp. 41-70.
- Antón, A., Potts, C. y Takahashi, K. (1994a). Inquiry-based scenario analysis of systems requirements. Technical report GIT-CC-94/14. Georgia Institute of Technology.
- Antón, A., Potts, C. y Takahashi, K. (1994b). Inquiry-based requirements analysis. *IEEE Software* 11(2), pp. 21-32.
- Barki, H., Rivard, S. y Talbot, J. (2001). An integrative contingency model of software project risk management. *Journal of Management Information Systems* 17(4), pp. 37-69.
- Behrens, H. (2002). *Requirements Analysis Using Statecharts and Generated Scenarios*. Doctoral Symposium at IEEE Joint Conference on Requirements Engineering. Essen, Germany.
- Byrd, T., Cossick, K. y Zmud, R. (1992). A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly* 16(1), pp. 117-138.
- Carroll, J. (2000). *Making use: Scenario-based design of human-computer interactions*. Massachusetts: MIT Press.
- Carroll, J. et al. (1998). Requirements development in scenario-based design. *IEEE transactions on Software Engineering* 24(12), pp. 1156-1170.
- Carroll, J. y Go, K. (2004). The blind men and the elephant: views of scenario-based system design. *Interactions* 11(6), pp. 44-53.
- Chatzoglou, P. y Macaulay, L. (1996). Requirements capture and IS methodologies. *Information Systems Journal* 6(3), pp. 209-225.
- Chen, C. et al. (1994). Formal approach to scenario analysis found. *IEEE Software* 11(2), pp. 33-34.
- Cheng, B. y Atlee, J. (2007). Research directions in Requirements Engineering. *Proceedings Future of Software Engineering* (pp. 285-303). Minneapolis, USA.
- Crowley, D. (1982). *Understanding communication: The signifying web*. New York: Gordon and Breach Science Publishers.
- Davis, A. y Hickey, A. (2003). Requirements elicitation and elicitation technique selection: A model for two knowledge-intensive software development processes. *Proceedings thirty-sixth annual Hawaii international conference on systems sciences* (pp. 1-10). Big Island, USA.
- Davis, G. (1982). Strategies for information requirements determination. *IBM Systems Journal* 21(1), pp. 4-31.
- do Prado, J. et al. (2007). Enhancing a requirements baseline with scenarios. *Requirements engineering* 2(4), pp. 184-198.

- de Roever, W. y Engelhardt, K. (1998). Data refinement: Model-oriented proof methods and their comparison. Cambridge: Cambridge University Press.
- Dubois, E. and Pohl, K. (2003). RE 02: A major step toward a mature requirements engineering community. *IEEE Software* 20(1), pp. 14-15.
- Duggan, E. y Thachenkary, C. (2004). Integrating nominal group technique and joint application development for improved systems requirements determination. *Information & Management* 41(4), pp. 399-411.
- Fliedl, G., Kop, C. y Mayr, H. (2005). From textual scenarios to a conceptual schema. *Data Knowledge Engineering* 55(1), pp. 20-37.
- Frolick, M. y Robichaux, B. (1995). EIS information requirements determination - Using a group support system to enhance the strategic business objectives method. *Decision Support Systems* 14(2), pp. 157-170.
- Gomaa, H. y Scott, D. (1981). Prototyping as a tool in the specification of user requirements. *Proceedings 5th international conference on Software Engineering* (pp. 333-342). San Diego, USA.
- Gottesdiener, E. (2008). Good practices for developing user requirements. *CrossTalk: The journal of defense Software Engineering*, March, pp. 13-17.
- Gunda, S. (2008). Requirements Engineering - Elicitation Techniques. Master's Thesis. Department of Technology, Mathematics and Computer Science, Software University West.
- Hickey, A. y Davis, A. (2004). A unified model of requirements elicitation. *Journal of Management Information Systems* 20(4), pp. 65-84.
- Holbrook, H. (1990). A scenario-based methodology for conducting requirements elicitation. *Software Engineering Notes* 15(1), pp. 96-104. I
- Holbrook, H. (1990). A scenario-based methodology for conducting requirements elicitation. *SIGSOFT Software Engineering Notes* 15(1), pp. 95-104.
- Hooper, J. y Hsia, P. (1982). Scenario-based prototyping for requirements identification. *Proceedings of the Workshop on Rapid Prototyping* (pp. 88-93). Columbia, USA.
- Iivari, J. (1992). The organizational fit of information systems. *Journal of Information Systems* 2(1), pp. 3-29.
- Jayaratna, N. (1994). Understanding and evaluating methodologies. London: McGraw Hill.
- Keil, M. y Carmel, E. (1995). Customer-developer links in software development. *Communications of the ACM* 38(5), pp. 33-44.
- Khedri, R. (2003). Requirements scenarios formalization technique: *N* versions towards one good version. *Electronic Notes in Theoretical Computer Science* 44(3), pp. 112-135.
- Kickert, W. (1983). Research note: Research models underlying situational dependency. *Organization Studies* 4(1), pp. 55-72.
- Kotonya, G. y Somerville, I. (2006). Requirements engineering process and techniques. USA: Willey.
- Kyng, M. (1995a). Creating contexts for design. In Carroll, J. (Ed.), *Scenario-based design: Envisioning work and technology in system development* (pp. 85-108). New York: Willey.
- Kyng, M. (1995b). Making representations work. *Communications of the ACM* 38 (9), pp. 46-55.
- Lamb, R. y Kling, R. (2003). Reconceptualizing users as social actors in information systems research. *MIS Quarterly* 27(2), pp. 197-235.
- Lyytinen, K. (1988). Expectation failure concept and systems analysts' view of information system failures: Results of an exploratory study. *Information & Management* 14(1), pp. 45-56.

- Lyytinen, K., Mathiassen, L. y Ropponen, J. (1998). Attention shaping and software risk - A categorical analysis of four classical risk management approaches. *Information Systems Research* 9(3), pp. 233-255.
- Markus, L. y Mao, J. (2004). Participation in development and implementation - Updating an old, tired concept for today's contexts. *Journal of the Association for Information Systems* 5(11-12), pp. 514-544.
- Mathiassen, L. (2007). A contingency model for requirements development. *Journal of the Association for Information Systems* 8(11), Article 33.
- McFarlan, F. (1981). Portfolio approach to information systems. *Harvard Business Review* 59(5), pp. 142-150.
- McFarlan, F. (1982). Portfolio approach to information systems. *Journal of Systems Management* 33(1), pp. 12-19.
- McKeen, J., Guimaraes, T. y Wetherbe, J. (1994). The relationship between user participation and user satisfaction - An investigation of 4 contingency factors. *MIS Quarterly* 18(4), pp. 427-451.
- Neill, C. y Laplante, P. (2003). Requirements Engineering: The state of the practice. *IEEE Software* 20(6), pp. 40-45.
- Nguyen, L. y Swatman, P. (2003). Managing the requirements engineering process. *Requirements Engineering Journal* 8(1), pp. 55-68.
- Peffer, K., Gengler, C. y Tuunanen, T. (2003). Extending critical success factors methodology to facilitate broadly participative information systems planning. *Journal of Management Information Systems* 20(1), pp. 51-85.
- Pfleeger, S. (1998). *Software Engineering - Theory and practices*. New York: Prentice Hall.
- Pohl, K. (1994). The three dimensions of requirements engineering - A framework and its applications. *Information Systems* 19(3), pp. 243-258.
- Pohl, K. (1997). *Requirements Engineering - An overview, encyclopedia of computer science and technology* (Vol. 36). New York: Marcel Dekker.
- Potts, C. (1995). Using schematic scenarios to understand user needs. *Proceedings 1st conference on designing interactive systems: Processes, practices, methods, & techniques* (pp. 247-256). Michigan, USA.
- Serna, M.E. (2012). Analysis and selection to requirements elicitation techniques. *Proceedings 7th Computing Colombian Congress* (pp. 1-7). Medellin, Colombia.
- Serna, M.E. y Serna, A.A. (2015). MoDeMaRE: Model to Development and Management Requirements Engineering. *IEEE Transactions on Software Engineering*. In press.
- Sutcliffe, A. (1998). Scenario-based requirements analysis. *Requirements Engineering* 3(1), pp. 48-65.
- Sutcliffe, A. (2003). Scenario-based Requirements Engineering. *Proceedings 11th IEEE International Conference on Requirements Engineering* (pp. 320-326). California, USA.
- Sutcliffe, A. (2003). Scenario-based Requirements Engineering. *Proceedings 11th IEEE International Requirements Engineering Conference* (pp. 320-329). Monterey Bay, USA.
- Vanderdonck, J. (2005). A MDA-compliant environment for developing user interfaces of Information Systems. *Lecture Notes in Computer Science* 3520, pp. 16-31.
- Webster, J. y Watson, R. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS Quarterly* 26(2), pp. xiii-xx.
- Weill, P. y Olson, M. (1989). An assessment of the contingency theory of management information systems. *Journal of Management Information Systems* 6(1), pp. 59-85.

- Wexelblat, A. (1987). Report on scenario technology. MCC technical report STP-139-87. Microelectronics and computer Technology Corporation. Austin: USA.
- Young, R. y Barnard, P. (1987). The use of scenarios in human-computer interaction research - Turbocharging the tortoise of cumulative science. ACM SIGCHI Bulletin 17(SI), pp. 291-296.
- Zhang, P. y Li, N. (2005). The intellectual development of human-computer interaction research: A critical assessment of the MIS literature (1990-2002). Journal of the Association for Information Systems 6(1), pp. 227-292.
- Zhang, Z. (2007). Effective requirements development - A comparison of requirements elicitation techniques. In Berki, E. et al. (Eds.), Software quality management XV: Software quality in the knowledge society (pp. 225-240). London: British Computer Society.

 Ángela María Tascon Calvo Estudiante	 Francisco Orlando Domínguez H. Estudiante
 Prof. Edgar Serna M. Asesor	
FECHA ENTREGA: _____	

_____ Comité trabajo de grado de la Facultad		
ACEPTADO ____	RECHAZADO ____	ACEPTADO CON MODIFICACIONES ____
ACTA NO. _____		
FECHA ENTREGA: _____		

_____ Consejo de Facultad	
ACTA NO. _____	
FECHA ENTREGA: _____	